

MPEG-4 sequence, content based object coding
MPEG-4 VOP time sequence

FIG. 1

```

for (all MBs) %process all MB form P-VOP%
    fullpel_motion_estimation(current MB from P-VOP on previous VOP)
for (all MBs) %process all MB form P-VOP%
    halfpel_motion_estimation(current MB from P-VOP on previous VOP)
for (all MBs) %process all MB form P-VOP%
    motion_compensation(current MB from P-VOP on previous VOP)
error_VOP = original_VOP - compensated_VOP
for (all MBs) %process all MB form P-VOP%
    code(current error MB from P-VOP)
for (all B-VOPs)
    for (all MBs) % process all MB form B-VOP%
        fullpel_motion_estimation(current MB from current B-VOP on previous VOP)
    for (all MBs) %process all MB from B-VOP%
        halfpel_motion_estimation(current MB from current B-VOP on previous VOP)
    for (all MBs) %process all MB form B-VOP%
        fullpel_motion_estimation(current MB from current B-VOP on next VOP)
    for (all MBs) % process all MB from B-VOP%
        halfpel_motion_estimation(current MB from current B-VOP on next VOP)
    for (all MBs) %process all MB from B-VOP%
        motion_compensation(current MB)
error_VOP = original_VOP - compensated_VOP
for (all MBs) %process all MB form B-VOP%
    code(current MB form Current B-VOP)

```

Original (VOP-oriented) MPEG-4 like source code

FIG. 2

```

for (all Mbs) % process all MB from P-VOP %
    fullpel_motion_estimation(current MB from P-VOP on previous VOP)
    halfpel_motion_estimation(current MB from P-VOP on previous VOP)
    motion_compensation(current MB from P-VOP)
    error_MP = original_MB-compensated_MB
    code(current error MB from P-VOP)
for (all B-VOPs)
    for (all Mbs) %process all MB from P-VOP%
        fullpel_motion_estimation(current MB from current B-VOP on previous VOP)
        halfpel_motion_estimation(current MB from current B_VOP on previous VOP)
        fullpel_motion_estimation(current MB from current B-VOP on next VOP)
        halfpel_motion_estimation(current MB from current B-VOP on next VOP)
        motion_compensation(current MB)
        error_MB = original_MB - compensated_MB
        code(current MB from current B-VOP)

```

Invented (MB-oriented) MPEG-4 like source code

FIG. 3

```
for (all MBs)
  full_motion_estimation(on previous VOP)
  halfpel_motion_estimation(on previous VOP)
  motion_compensation(on previous VOP)
for (all MBs)
  fullpel_motion_estimation(on next VOP)
  halfpel_motion_estimation(on next VOP)
  motion_compensation(on next VOP)
  interpolation_mode(best match previous and next VOP)
```

Original (MB-oriented) MPEG-4 like source code
for B-VOP encoding in interpolated mode.

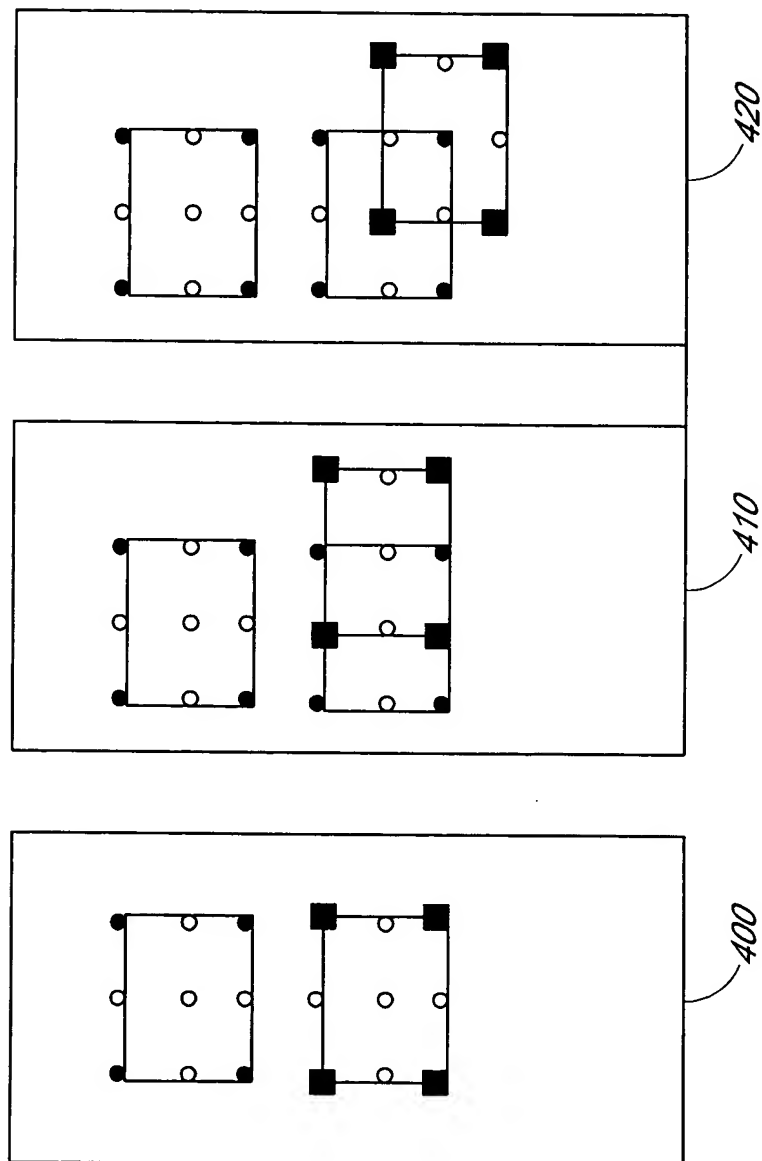
FIG. 4

5 / 24

```
for (all MBs)
  fullpel_motion_estimation(on previous VOP)
for (all MBs)
  fullpel_motion_estimation(on next VOP)
  halfpel_motion_estimation(on next VOP)
  halfpel_motion_estimation(on previous VOP)
  motion_compensation(on previous VOP)
  motion_compensation(on next VOP)
  interpolation_mode(best match previous and next VOP)
```

Invented (MB-oriented) MPEG-4 like source code
for B-VOP encoding in interpolated mode.

FIG. 5



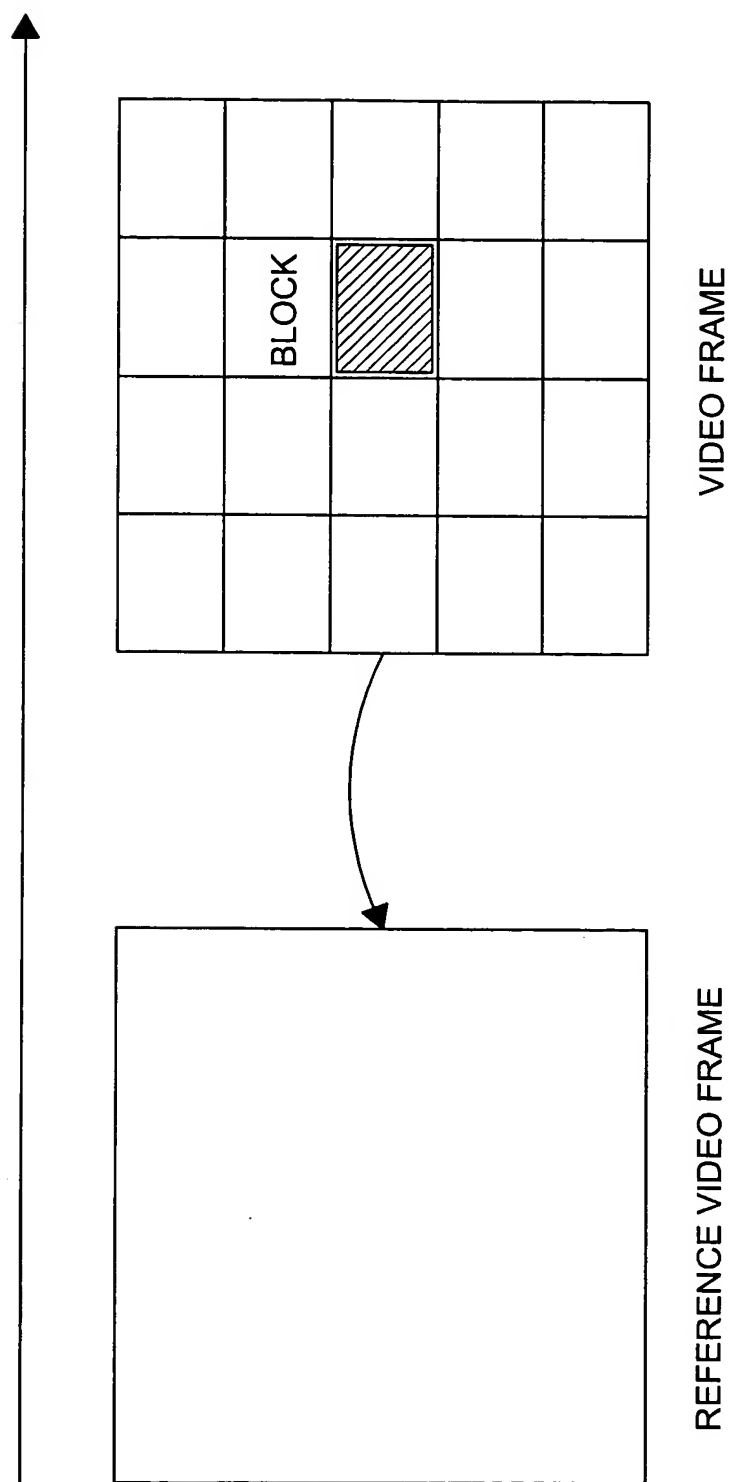


FIG. 7

8 / 24

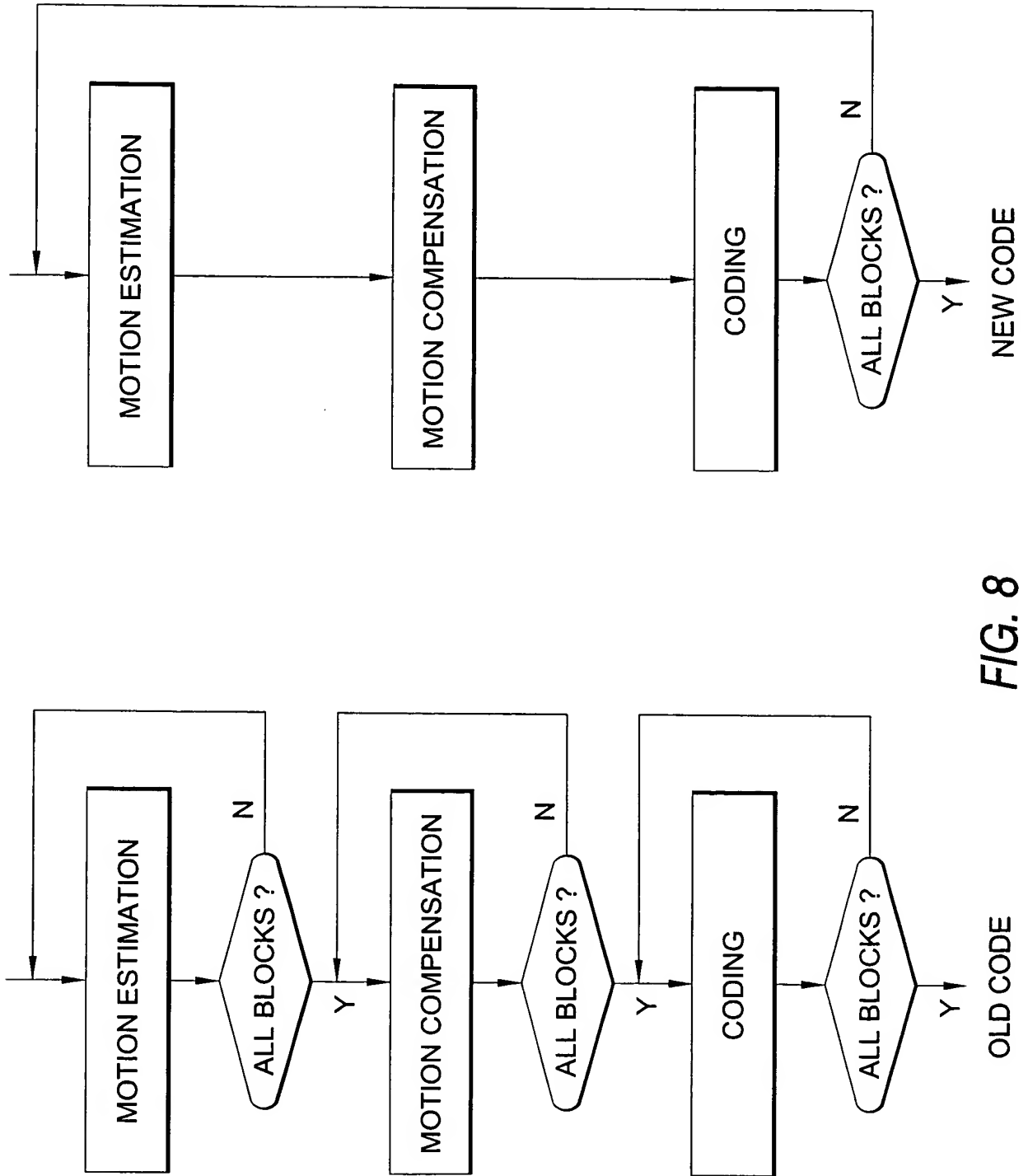


FIG. 8

9 / 24

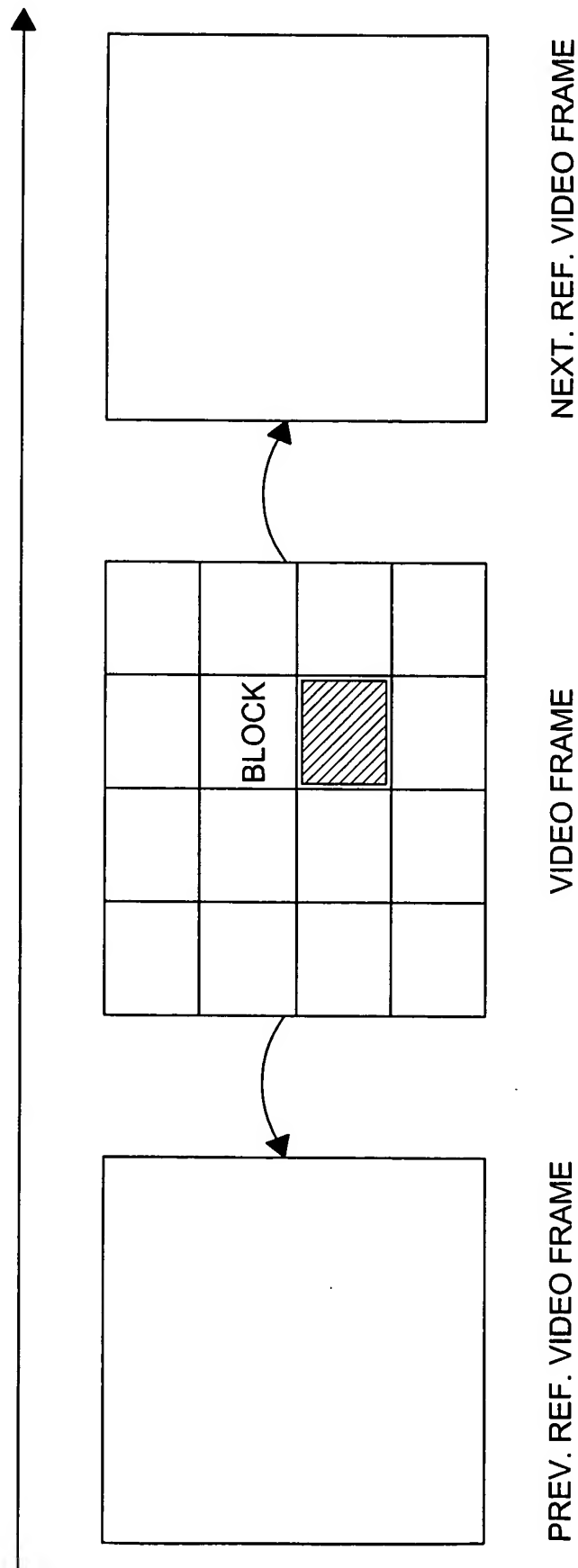


FIG. 9

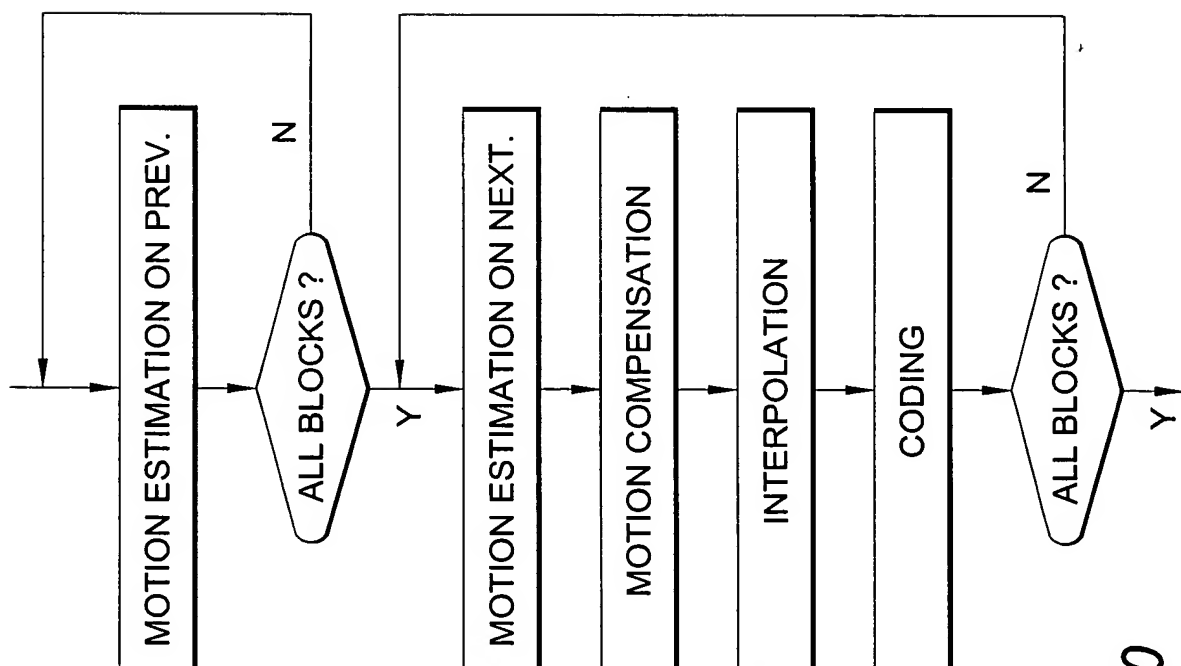


FIG. 10

11 / 24

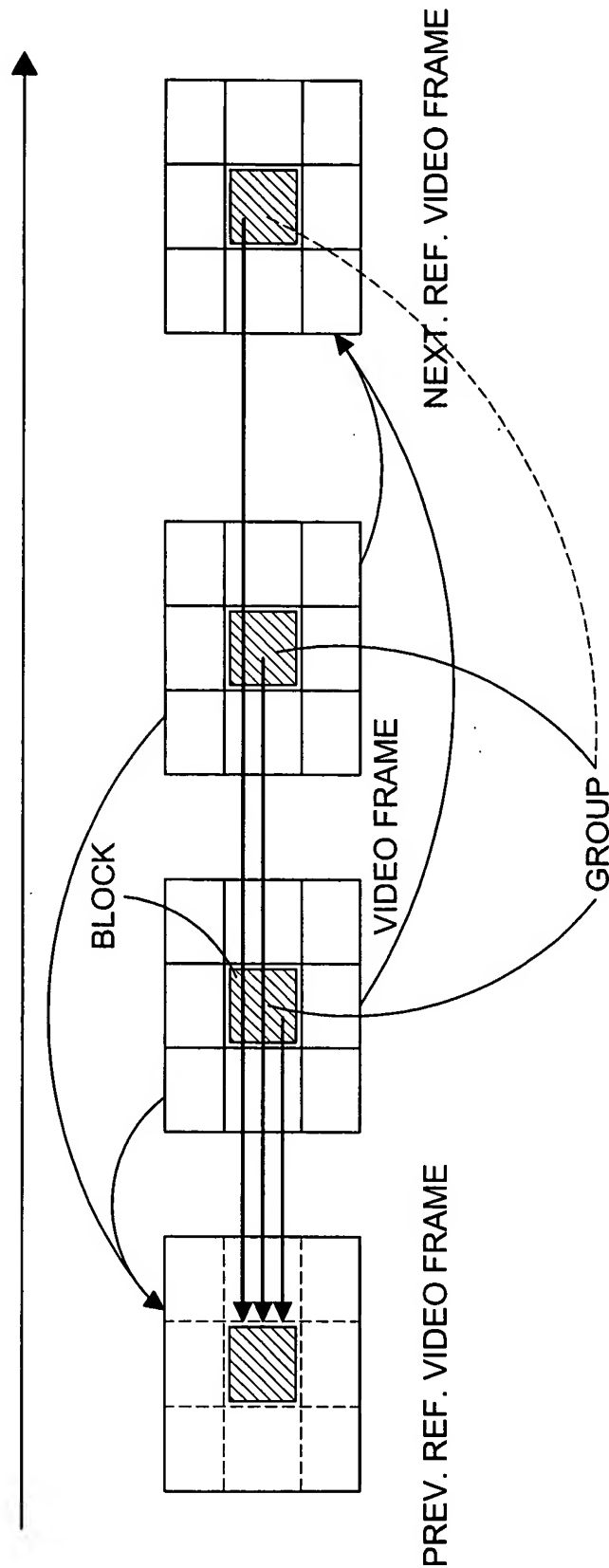


FIG. 11

12 / 24

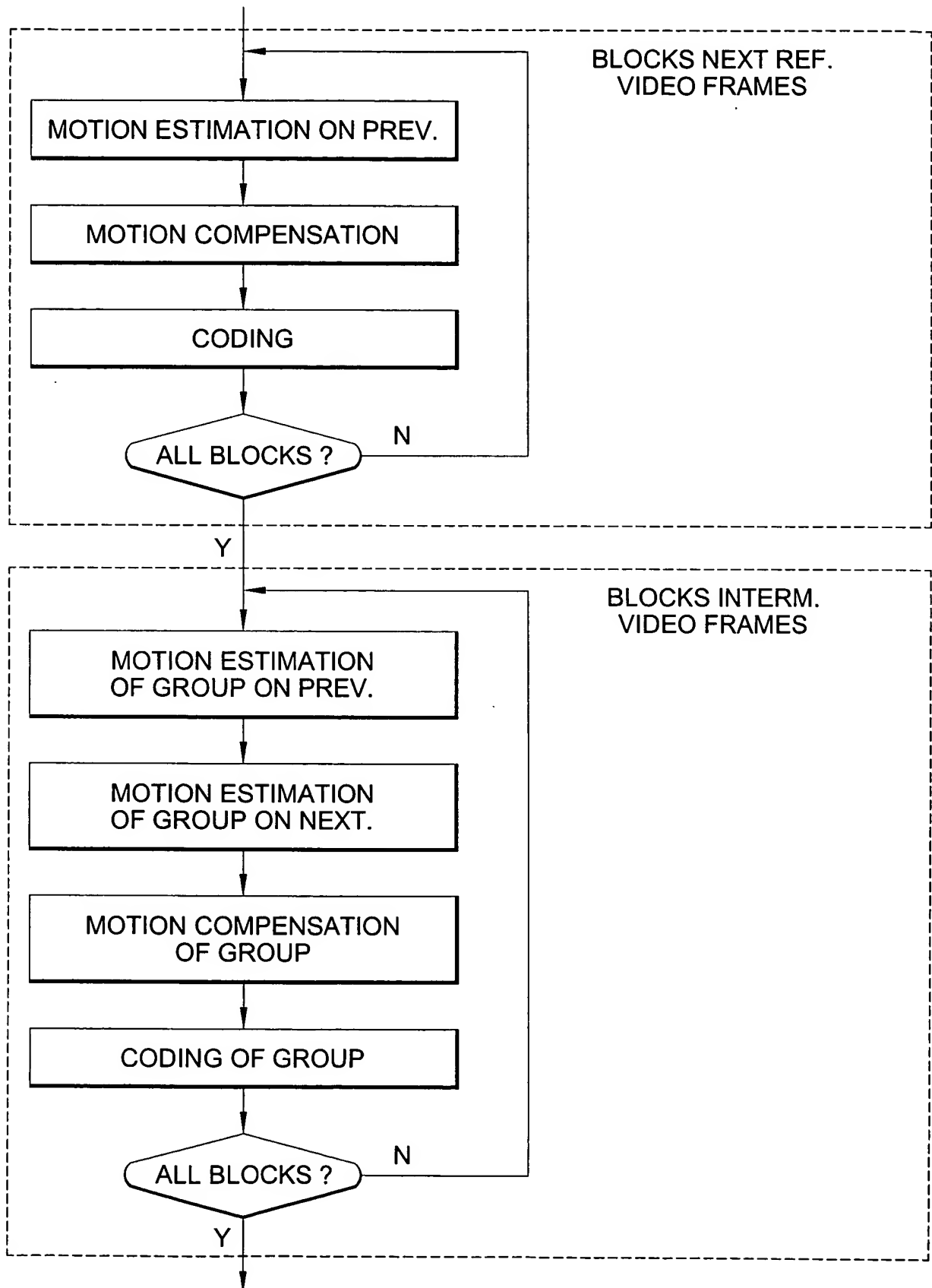


FIG. 12

13 / 24

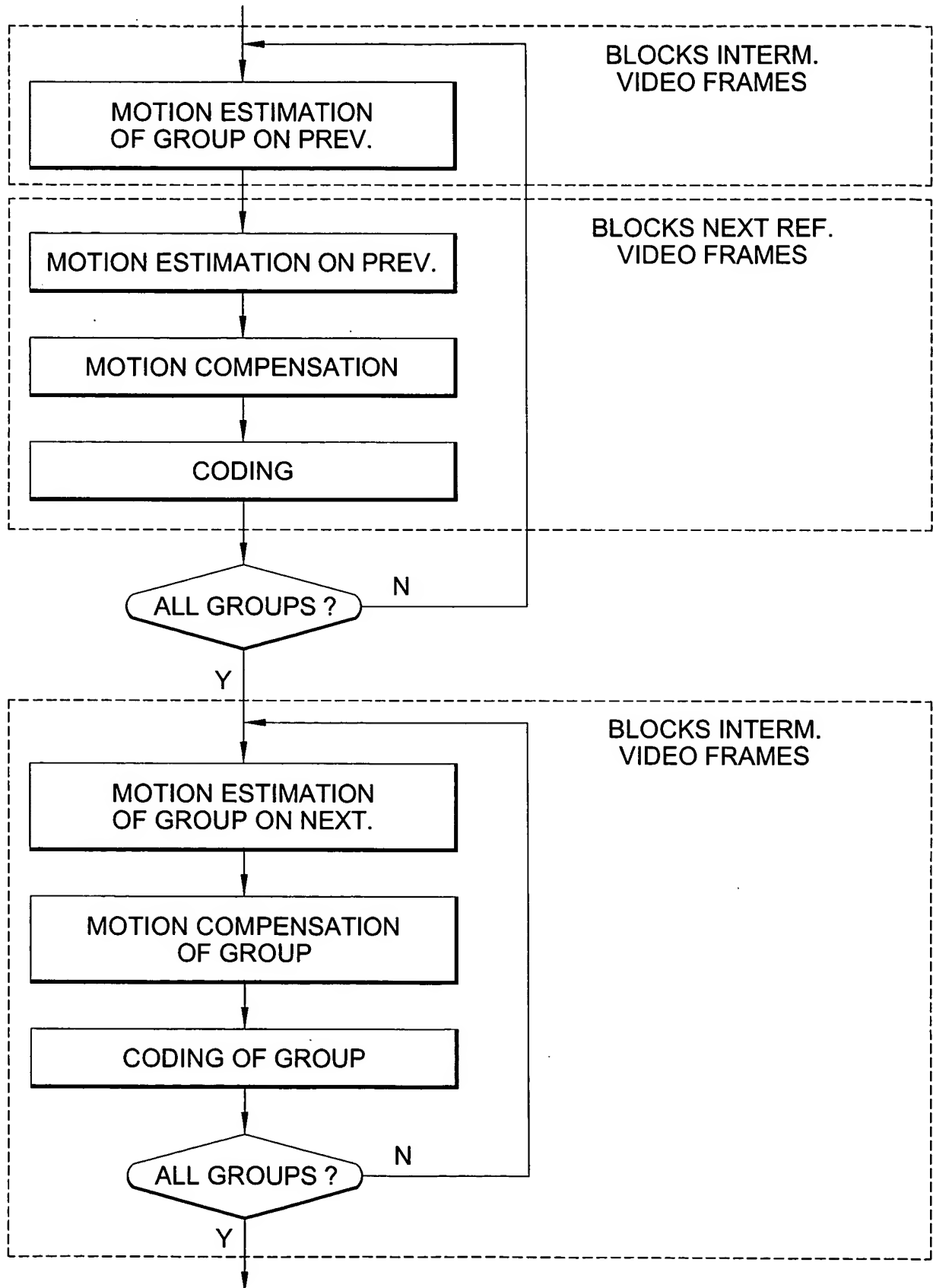


FIG. 13

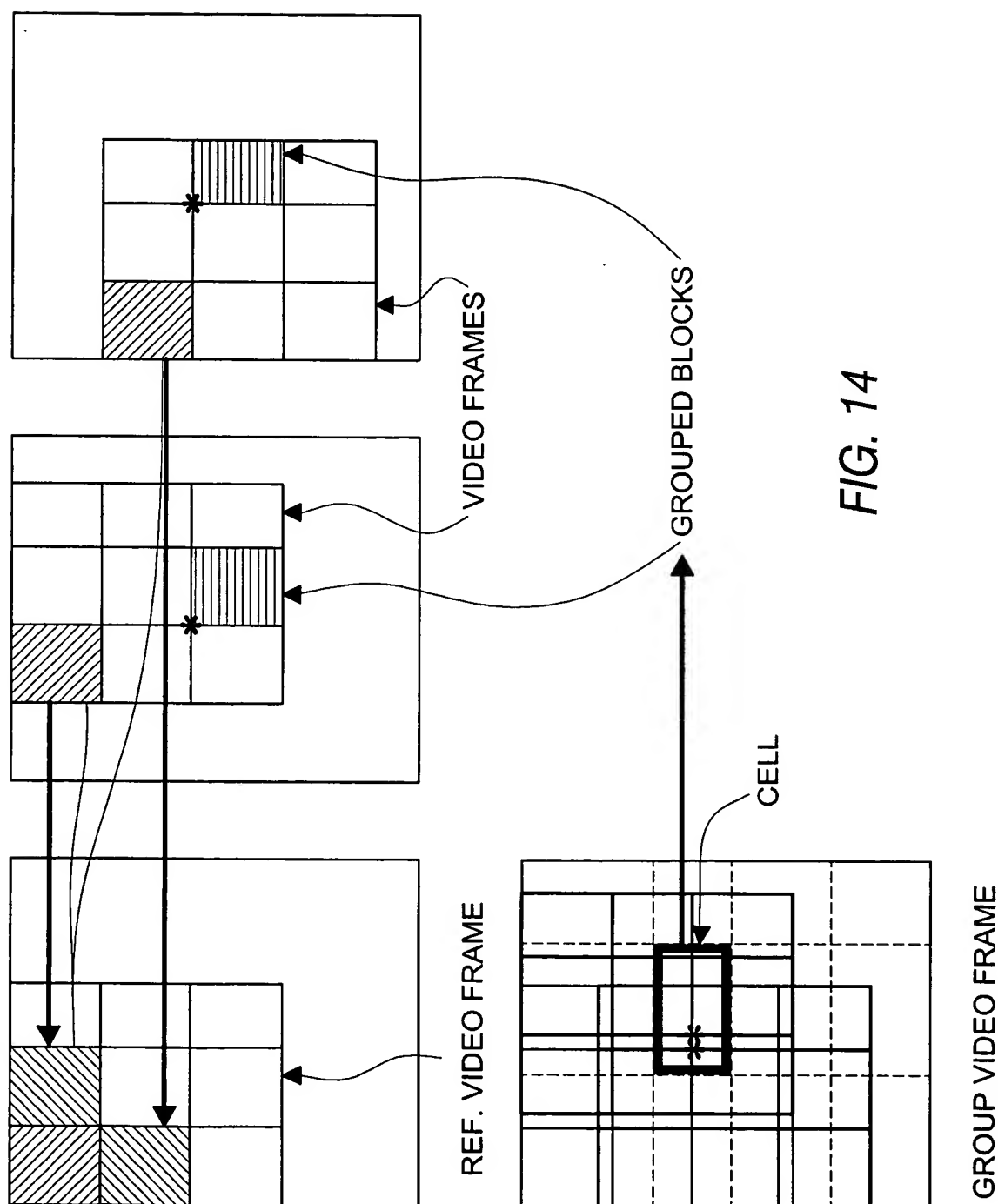


FIG. 14

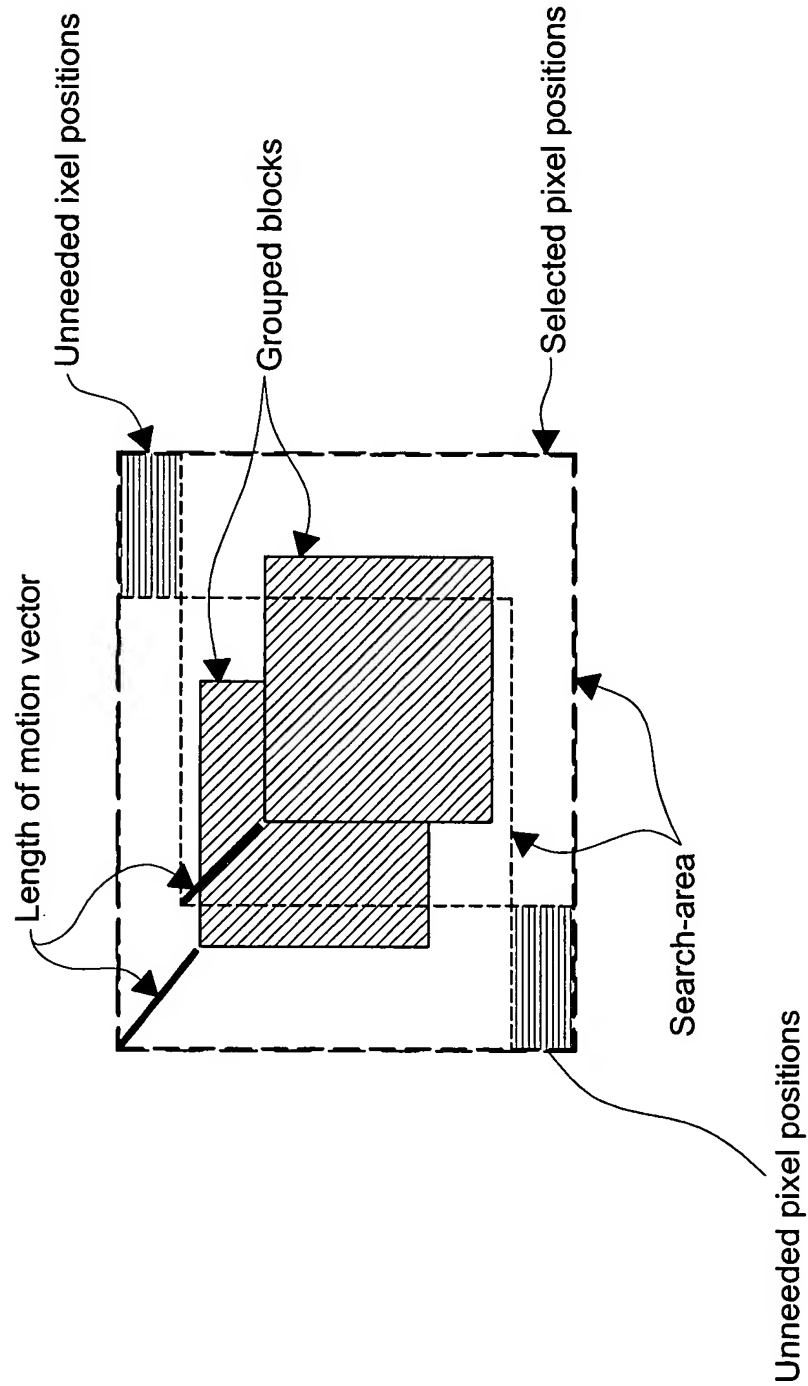


FIG. 15

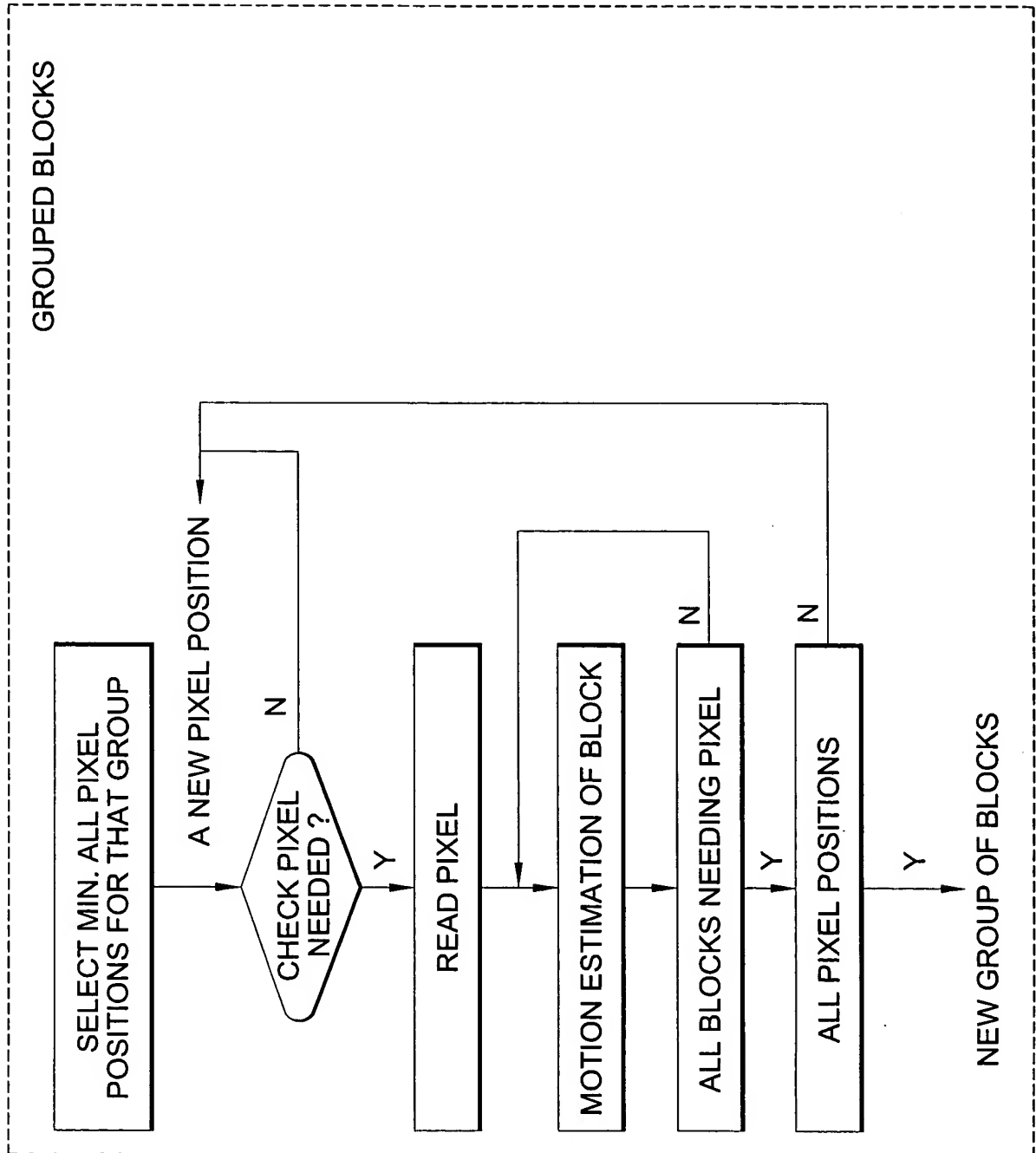


FIG. 16

17 / 24

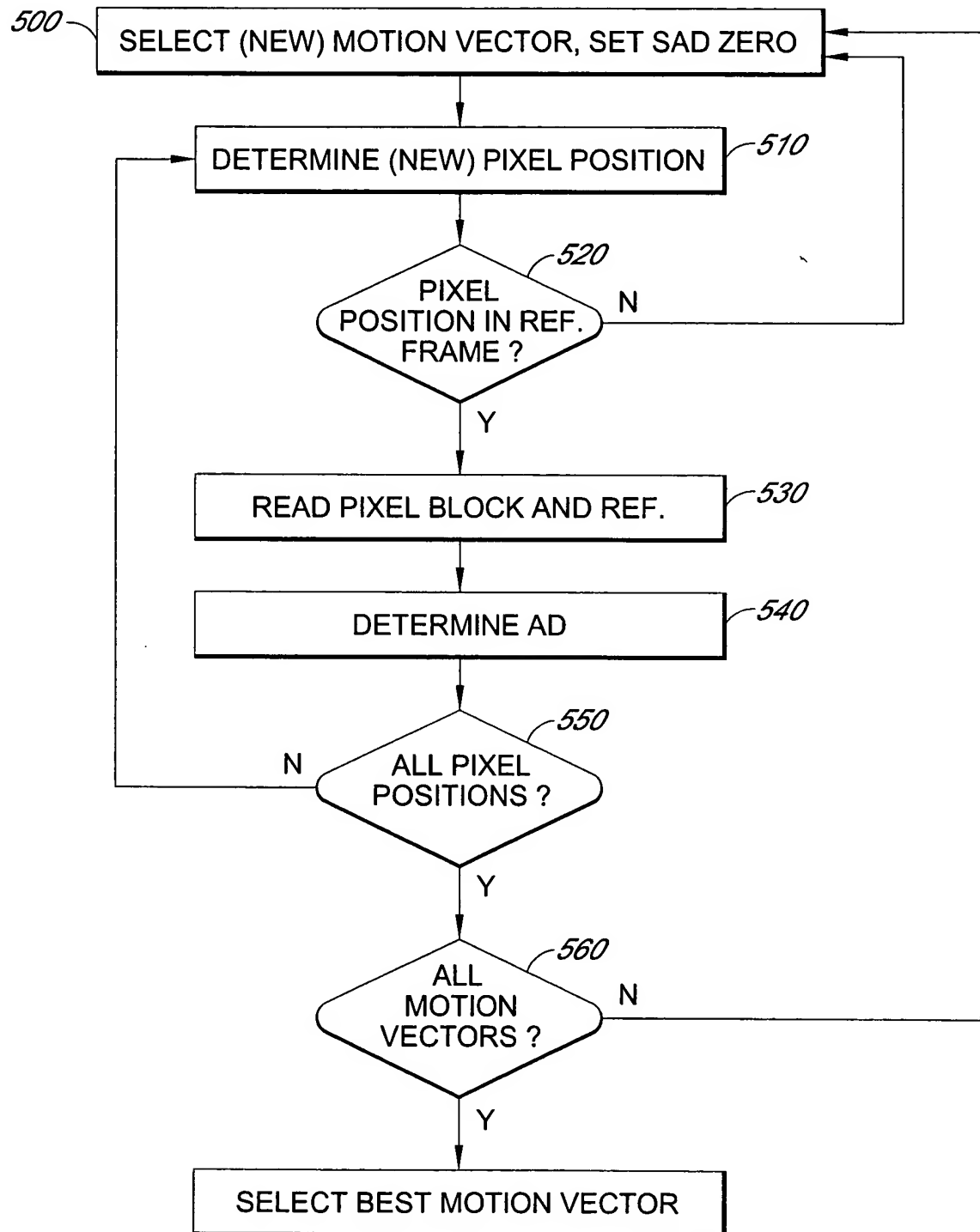


FIG. 17

18 / 24

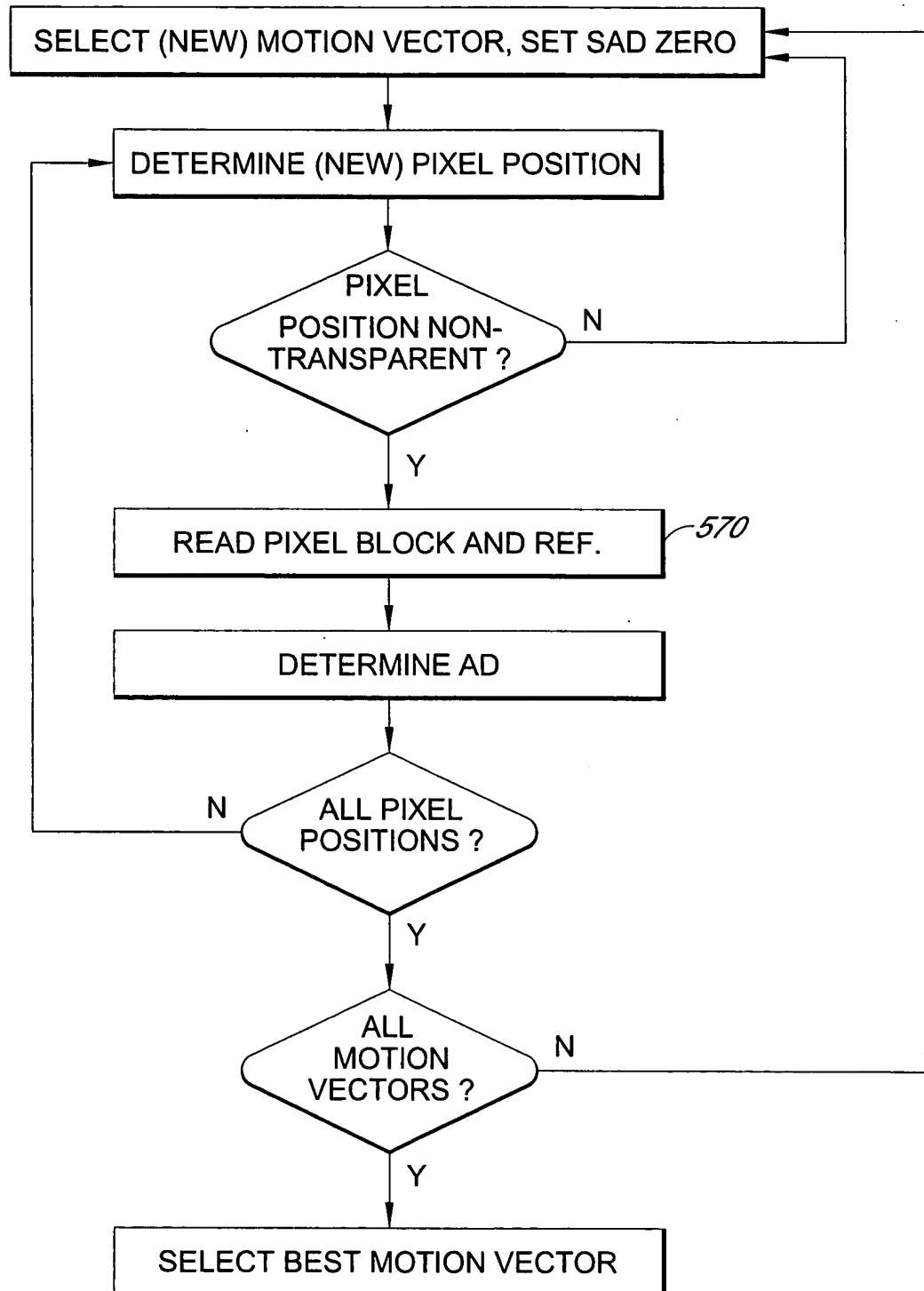


FIG. 18

19 / 24

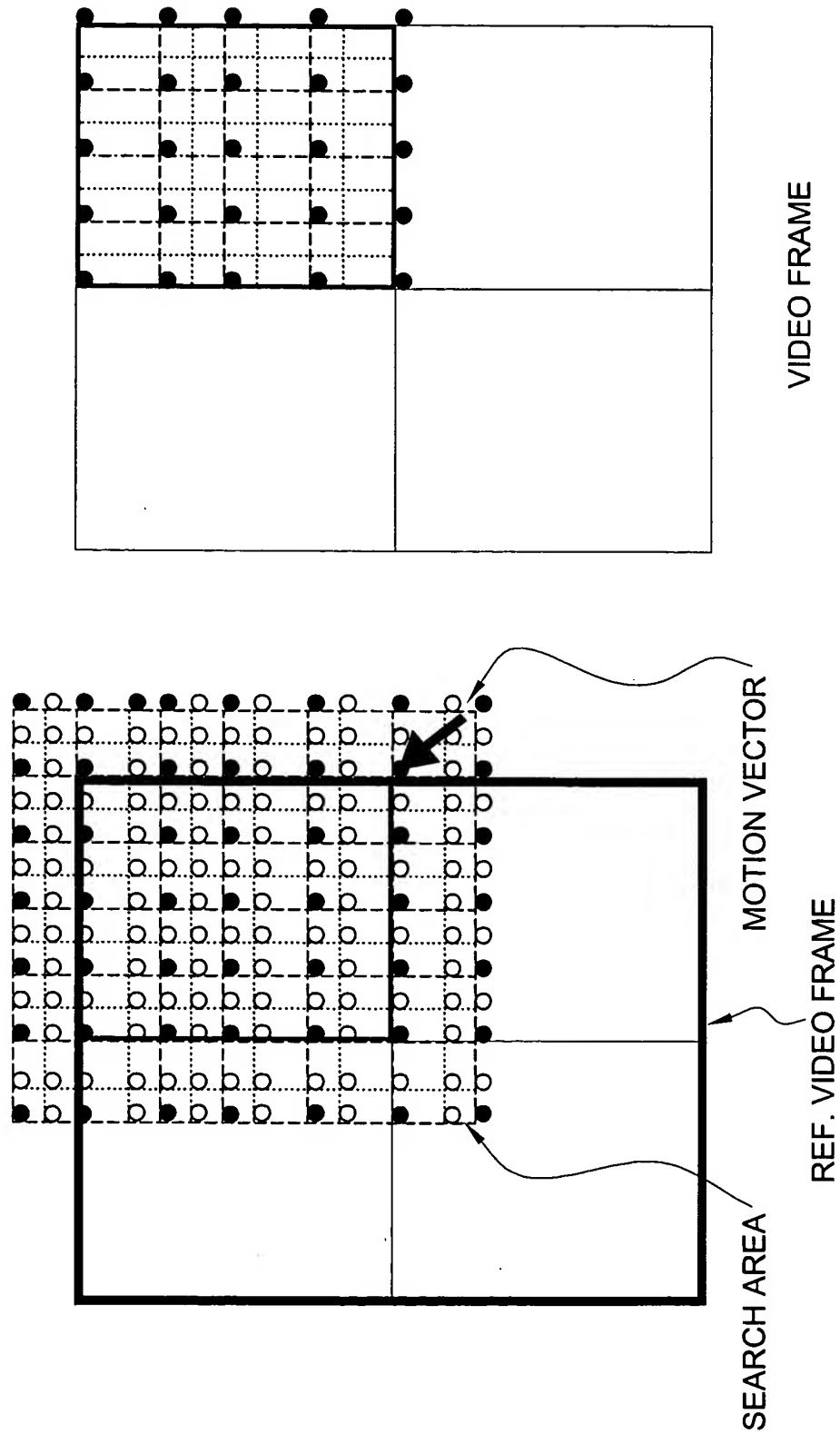


FIG. 19

20 / 24

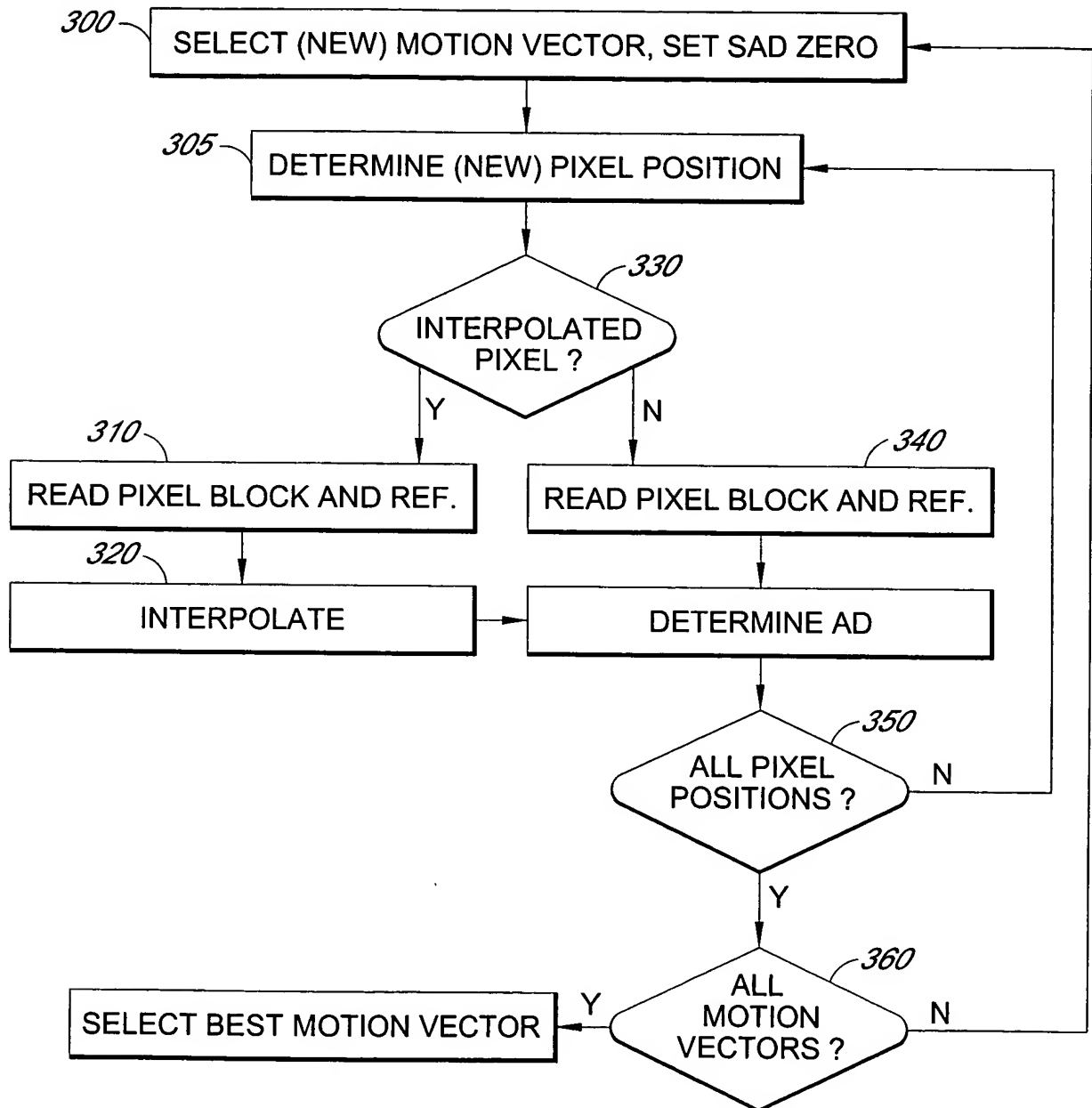


FIG. 20

21 / 24

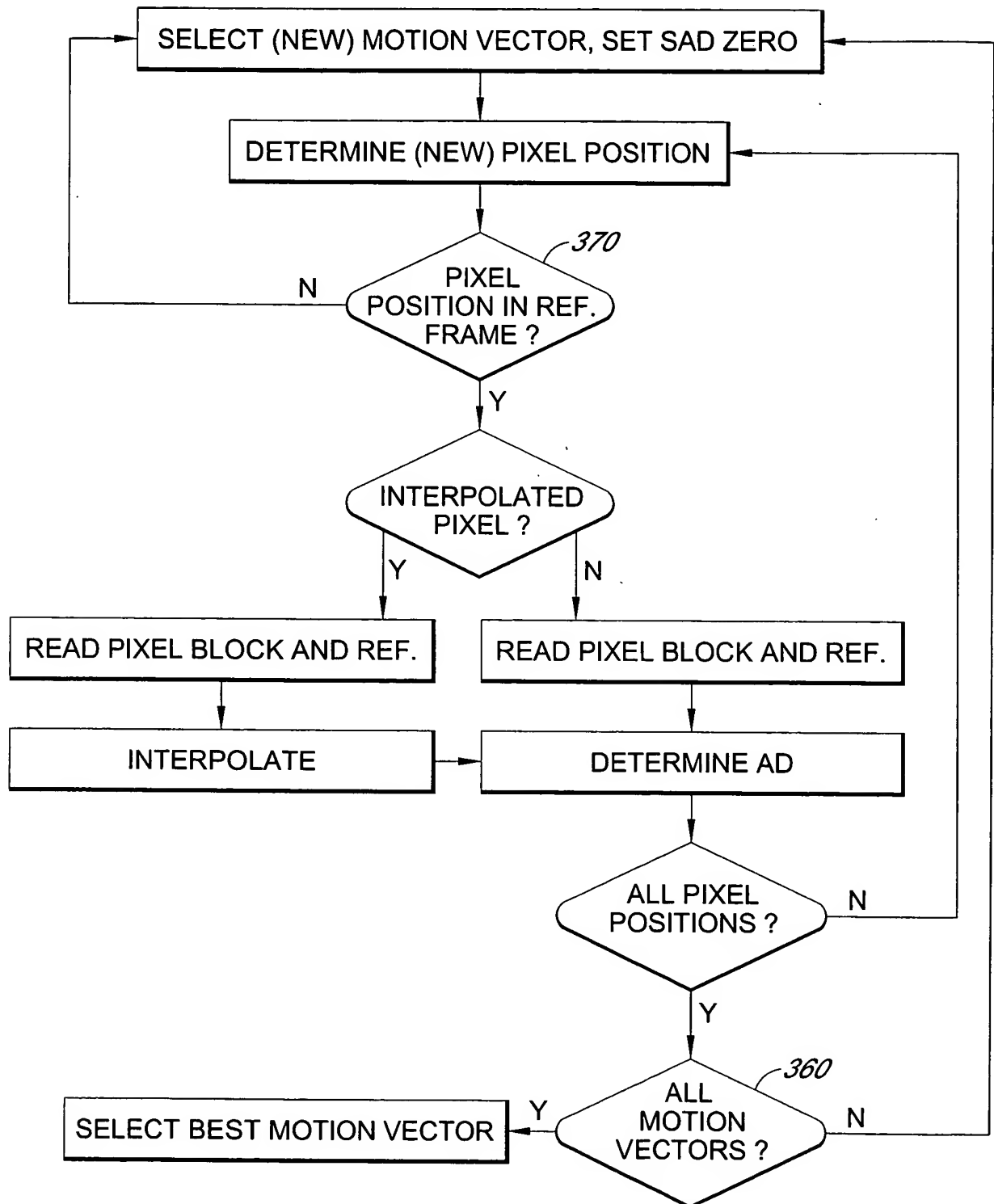


FIG. 21

22 / 24

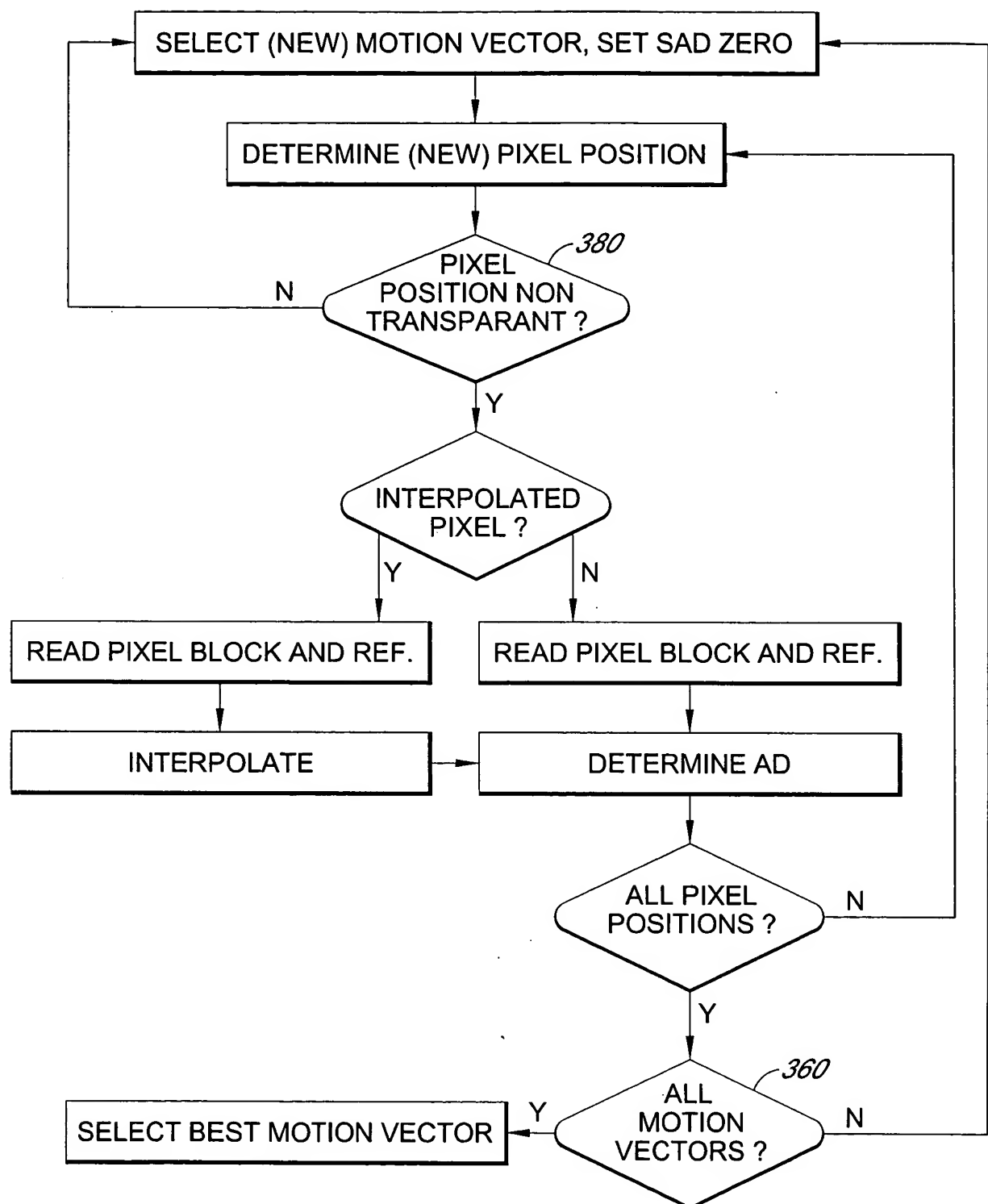
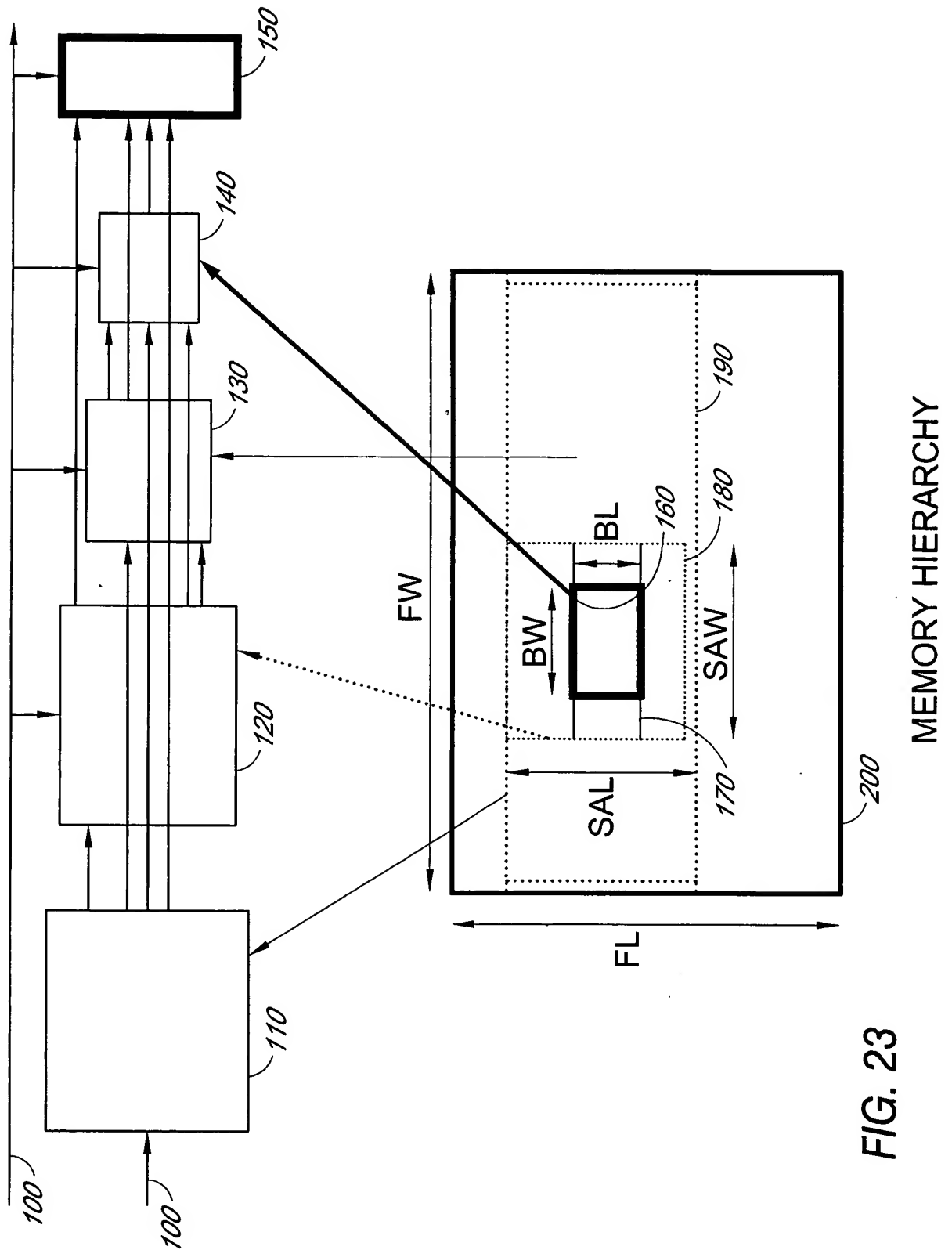
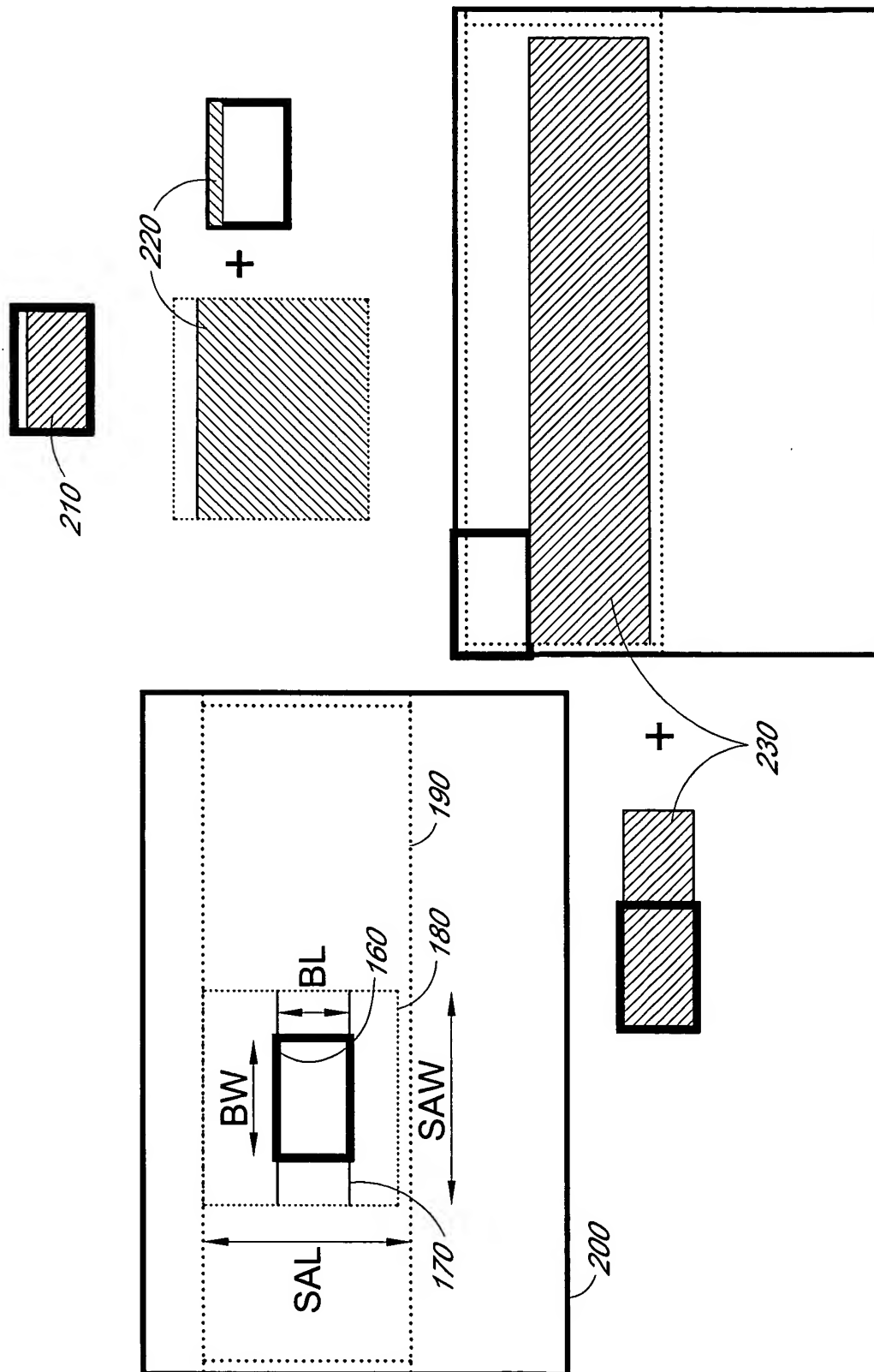


FIG. 22





MEMORY SIZES

FIG. 24